* مفردات لغة C++ :

أي برنامج يستخدم ضمنه عمليات الإدخال والإخراج يجب تضمين الملف الرأسي <iostream.h> الحاوي على عمليات الإدخال والإخراج بالشكل التالي: #include <iostream.h>

ويجب على أي برنامج أن ينتهي بالتعليمة return 0 التي تعني أن التابع هو تابع رئيسي ولا يعيد أي قيمة

* التصريح عن المتغيرات:

يمكن التصريح عن المتغيرات في أي مكان ضمن البرنامج شرط قبل استخدام المتحول

ويجب التصريح عن جميع المتغيرات المستخدمة ضمن البرنامج بالشكل

نمط المعطيات   اسم المتحول ;

int x;

ويجب على أية تعليمة أن تنتهي بفاصلة منقوطة مثال

وإذا كان لدينا مجموعة متغيرات من نفس النمط فإنه يمكن أن نستخدم إحدى الصيغتين:

```
int x;
int y;   ⟹   int x, y, z;
int z;
```

كما يمكن التصريح عن المتغيرات وإسنادها في نفس الوقت مثال: int x=5;

* تعليمة الطباعة: نستخدم إما الطباعة عبارة أو طباعة قيمة

مثال: طباعة عبارة: cout<<" Hellow";

طباعة قيمة: cout<<"x=" << x;

* سلاسل الهروب: توجد مجموعة من سلاسل الهروب ضمن لغة C++ وهي:

① \n نستخدم للانتقال إلى بداية سطر جديد

```
Cout<<"Welcome \n to C++";
Welcome
to C++
```

② \t نستخدم لترك فراغ (tab المسافة معلومة)

```
Cout<<"Welcome to \t C++";
Welcome to        C++
```

④ \a لقرع جرس النظام (alret)

⑤ \\ و \" لطباعة الرمز \ والرمز " على الترتيب

* تعليمة الاسناد: تستخدم تعليمة الاسناد = لاسناد القيمة الموجودة على يمينها الى القيمة الموجودة على يسارها وتستخدم لاسناد قيمة (عددا او حرف ......) x = 2 ;

او لاسناد نتيجة تابع X = Pow(2, 3);

او قيمة متحول x = 3;

y = x;

* العمليات الحسابية: وهي + , - , * , / , %

بالنسبة لعملية القسمة اذا كان معاملاها عددين صحيحين يكون الناتج عدد صحيح والا يكون الناتج عدد حقيقي

8/2 = 4

9/2 = 4

9.0/2 = 4.5

* تعليمات الاسناد المختصرة:

توجد ضمن لغة C++ مجموعة من عمليات الاسناد المختصرة وهي:

+= , -= , *= , /= , %=

تستخدم لاخذ القيمة الموجودة على يسار عملية الاسناد وتطبيق العملية مع استخدام القيمة الموجودة على يمينها وتخزين الناتج في المتحول الموجود على يسارها

```
int x = 2;
X += 4;   →  X = X+4  ⇒ X=6
X -= 1;   →  X = X-1  ⇒ X=5
X *= 2;   →  X = X*2  ⇒ X=10
X %= 5;   →  X = X%5  ⇒ X=0
```

* تعليمة الزيادة بواحد ++

تستخدم لزيادة قيمة المتحول بمقدار واحد وهي نوعين عملية الزيادة بواحد امامية وعملية الزيادة بواحد خلفية

الاولى: تقوم بالزيادة ثم التنفيذ

الثانية: تقوم بالتنفيذ ثم الزيادة

ونفس الكلام بالنسبة لعملية الانقاص --

مثال:

```
#include <iostream.h>
main()
{ int x = 10;
  Cout << x << "\n";     → 10
  Cout << ++x << "\n";   → 11
  Cout << x << "\n";     → 11
```

```
X = 10;
Cout << X << "\n";        → 10
Cout << X ++ << "\n";     → 10
Cout << X << "\n";        → 11
}
return 0;
```

مثال:

```
int x = 2;   int y = 1;
Z = 3 + 5 * X + (++y - 5) + 9
Z = 3 + 5 * X + (-3) + 9
Z = 3 + 10 - 3 + 9
  = 13 - 3 + 9 = 10 + 9 = 19
Z = 19
```

مثال:

```
X = 2;  y = 3;
Z = ++y - 5 + 3 * (y++) - 3 * x;
Z = 4 - 5 + 10 - 6
```

* أولويات العمليات: يتم تحديد الأولويات وفق ما يلي:

1- إشارة العدد
2- الضرب والقسمة
3- الجمع والطرح

وفي حال وجود عمليتين لهما نفس الأولوية يتم تنفيذ العملية التي على اليسار أولاً وأي عملية بين قوسين لها الأولوية أولاً.

* تعليمة الإدخال: cin عند وجودها يتم ايقاف البرنامج عن التنفيذ حتى يتم قراءة متغير من لوحة المفاتيح ويمكننا استخدام التعليمة cin للإدخال لأكثر من متغير بنفس الوقت وليس بالضرورة أن يكون لهذه المتغيرات نفس النمط

```
int x, y, z;                 int x, y, z;
Cin >> x >> y >> z;    ⇒     Cin >> x;
                             Cin >> y;
                             Cin >> z;
```

مثال:

```
int x;
float y;
bool z;
Cin >> x >> y >> z;
```

* عبارة if الشرطية : الصيغة العامة لها

```
if (condition)
{ statements }
```

معنى العبارة السابقة : اذا كان الشرط المنطقي محقق نفذ العمليات ، واذا كان الشرط غير محقق لا يتم تنفيذ شيء (لا يتم تنفيذ اي من العمليات التالية التابعة لـ if)

* عبارة if - else : الصيغة العامة لها

```
if (Condition)
 { statement1 ; }
else
 { statement2 ; }
```

معنى العبارة : اذا كان الشرط المنطقي محقق نفذ العملية او العمليات ① والا نفذ العملية او العمليات ②

مثال : اكتب برنامج يقوم بحساب مساحة ومحيط الدائرة

```
#include <iostream.h>
main()
{ float r,s,p;
Cout<<"r="; Cin>>r;
if(r>0)
 { s=3.14*r*r;
   p=2*3.14*r;
   Cout<<"s="<<s<<"\n";
   Cout<<"p="<<p<<"\n";
 }
else
 { Cout<<"error"; }
return 0;
}
```

MARIA ATIA

* اذا كانت العملية مركبة يجب حصرها بين قوسين

مثال : اكتب برنامج يقوم بحساب مساحة مثلث

$S_1=(a+b+c)/2$

$S=\sqrt{S_1(S_1-a)(S_1-b)(S_1-c)}$

```
#include <iostream.h>
#include <math.h>
main()
{ float a,b,c,s1,s;
Cout<<"a="; Cin>>a;
Cout<<"b="; Cin>>b;
Cout<<"c="; Cin>>c;
```

T T F

```
if ((a>0)&&(b>0)&&(c>0)&&(b+c>a)&&(b+a>c)
        &&(a+c>b))
{ S1=(a+b+c)/2;
  S=sqrt(S1*(S1-a)*(S1-b)*(S1-c));
  Cout<<"area="<<S;
}
else
   Cout<<"error";
return 0;
}
```

* ملاحظة: يستخدم العمليات المنطقية اي مع الشروط العمليات التالية:

> , < , >= , <= , == , != , && , || , !Not

```
true && true → true          true || true → true
true && false → false        true || false → true
false && true → false        false || true → true
false && false → false       false || false → false
!(true) → false
!(false) → true
```

اكتب برنامج يدخل المستخدم مرتبة الآحاد ويحسب البرنامج العدد الثلاثي علما ان العشرات ضعف الآحاد والمئات سبعة اضعاف الآحاد ومجموع الآحاد والعشرات والمئات مساوياً 10

```
#include <iostream.h>
main()
{ int x,y,z,t;
Cout<<"x="; Cin>>x;
 y=2*x;
 z=7*x;
if(x+y+z==10)
 t=x+y*10+z*100;
 Cout<< "t="<<t;
else
Cout<<"error";
return 0;
}
```

مثال: اكتب برنامج يدخل المستخدم ثلاثة منازل و يعطي العدد المكون منها:

```
#include <iostream.h>
main()
{int x,y,z,t;
 Cout<<"x="; Cin>>x;
 Cout<<"y="; Cin>>y;
 Cout<<"z="; Cin>>z;
if ( (x>=0) && (y>=0) && (z>=0))
 {  t= x + y*10 + z*100;
  Cout<<"t="<< t;
  }
else   Cout<<" there is not number";
}
```

اكتب برنامج يقوم باختبار فيما اذا كان X قاسم لـ Y:

```
#include <iostream.h>
main()
{ bool f = true;
  int x,y;
  Cout<<"X="; Cin>>x;
  Cout<<"y="; Cin>>y;
  if (y% x != 0)
   f = false;
  if ( f==true) أو  if(f)
  Cout<<" X is Kassem to y";
  else
  Cout <<" X is not Kassem to y";
  return 0;
 }
```

*عبارة if المتداخلة: الصيغة العامة لها

```
if (condition 1)
   if (condition 2)
      if (condition 3)
         statement 3
      else
         statement 33;
   else
      statement 22;
else
   statement 11;
```


اكتب برنامج لحل معادلة من الدرجة الأولى $ax+b=c$

$ax+b=c$

عدد لا نهائي $a=b=c=0$

مستحيل $a=b=0 \quad c\neq 0$

مستحيل $a=0 \quad b\neq 0$

$a\neq 0 \quad x=\frac{c-b}{a}$

```
#include<iostream.h>
main()
{ float a,b,c,x;
  Cout<<"a="; Cin>>a;
  Cout<<"b="; Cin>>b;
  Cout<<"c="; Cin>>c;
if(a==0)
    if(b==0)
     if(c==0)
        Cout<<"many solutions";
     else
        Cout<<"impossible";
    else
       Cout<<"impossible";
else
 { x=(c-b)/a;
   Cout<<"X="<<x;
 }
return 0;
}
```

اكتب برنامج لقياس حساب الزاوية بين متجهين :

$\vec{u} = u_1 \vec{i} + u_2 \vec{j} + u_3 \vec{K}$

$\vec{v} = v_1 \vec{i} + v_2 \vec{j} + v_3 \vec{K}$

$\cos\theta = (u_1 v_1 + u_2 v_2 + u_3 v_3) / |\vec{u}| . |\vec{v}|$

$|\vec{u}| = \sqrt{u_1^2 + u_2^2 + u_3^2}$

```
#include <iostream.h>
#include <math.h>
main()
{ float u1, u2, u3, v1, v2, v3, t, t2, t1 ;
  Cout<<" Enter v1, v2, v3, u1, u2, u3 " ;
  Cin>> u1>> u2>> u3>> v1>> v2>> v3 ;

  t1 = sqrt ((u1*u1) + (u2*u2) + (u3*u3)) ;   // طول الضلع (u)
  t2 = sqrt ((v1*v1) + (v2*v2) + (v3*v3)) ;   // طول (v)
  t = (u1*v1 + u2*v2 + u3*v3) / (t1*t3) ;
  if( t==0 )
   Cout<<" these vectors are cal " ;
  else
    Cout<<" these vectors are not cal " ;
  return 0 ;
 }
```

اكتب برنامج لاختبار فيما اذا كان المثلث الذي اضلاعه a, b, c قائم ام لا اعتمادا على فيثاغورث

```
#include <iostream.h>
main()
{ float a, b, c ;
  Cout<<" Enter a & b & c " ;
  Cin>> a >> b >> c ;
  if ((a>0) && (b>0) && (c>0) && (a+b>c) && (a+c>b) &&
       (b+c>a))
      if(( a*a + b*b == c*c) || (a*a + c*c == b*b) ||
           (b*b + c*c == a*a))
```

```
Cout<<" it is true" ;        // أي أنه قائم
else
 Cout<<" it's not true" ;    // غير قائم
else
 Cout<<" this shape isn't rectangle" ;   // ليس مثلث
return 0 ;
}
```

اكتب برنامج لإيجاد أكبر قيمة بين ثلاث أعداد صحيحة مدخلة من لوحة المفاتيح

```
#include <iostream.h>
main()
{ int x, y, z, max ;
 Cout<<"Enter the numbers" ;
 Cin>> x >> y >> z ;
 Max = x ;
 if (y > Max)
   Max = y ;
 if (z > Max)
    Max = z ;
 Cout<<"Max is" << Max ;
return 0 ;
}
```

حل 2

```
if (x > y)
  if (x > z)
    max = x ;
  else
    max = z ;
else
  if (y > z)
  max = y ;
Cout<<"max :" << max ;
return 0 ; }
```

اكتب برنامج لاختبار فيما إذا كان x مضاعف مشترك للعددين z و y :

```
#include <iostream.h>
main()
{ int x, y, z ;
  bool f ;
  Cout<<" Enter x & y & z" ;
  Cin>> x >> y >> z ;
```

```
f=false;
if ((x%y==0) && (x%z==0))
    f=true;
if(f)
  cout<<"it's true";
else
   cout<<"it's false";
return 0;
}
```

* الحلقات التكرارية : حلقة for

```
for ( exp1; exp2; exp3)
 { statement; }
```

مبدأ عمل الحلقة السابقة هو بالشكل التالي:

يتم البدء بالقيمة الابتدائية الموجودة عند exp1 ثم اختبار الشرط exp2 اذا كان محقق يتم تنفيذ التعليمة أو التعليمات statement ثم الانتقال إلى exp3 والقيام بالزيادة او النقصان ثم اختبار الشرط exp2 وهكذا...

يتم الخروج من الحلقة عندما يصبح الشرط exp2 غير محقق

واذا كانت التعليمة مركبة يتم حصرها بين قوسين

اكتب برنامج يقوم بحساب مجموع قواسم عدد ما.

```
#include <iostream.h>
main()
{  int x, s;   s=0;
  cout<<"enter x:"; cin>>x;
  for(int i=1; i<=x; i++)
      if( x%i==0)
         s=s+i;
  cout<<"sum="<<s;
return 0;
 }
```

*

اكتب برنامج لايجاد القاسم المشترك الأكبر بين عددين.

```
#include <iostream.h>
main ()
{ int a,b,c;
 cout<<"a="; cin>>a;
 cout<<"b="; cin>>b;
 for (int i=1; i<=a; i++)
   if ((a%i==0)&&(b%i==0))
     c=i;
     cout<<"c="<<c;
 return 0;
}
```

اكتب برنامج لايجاد المضاعف المشترك الأصغر لعددين
اذا كان العددين اوليين
عندها يكون المضاعف الأصغر لها
هو حاصل ضربهما
مثلاً 5 و 7 ⇐ المضاعف 35

```
 for(int i=1; i<=a*b; i++)
   if ((i%a==0)&&(i%b==0))
   { c=i;
      break;
   }
  cout<<"c="<<c;
 return 0;
}
```

اكتب برنامج يقوم بحساب العاملي لكل عدد زوجي ضمن المجال [1, x]

```
#include <iostream.h>
main(){ int x,f;
  cout<<"x="; cin>>x;
  for(int i=1; i<=x; i++)
     if (i%2==0)
     { f=1;
        for(int j=1;j<=i;j++)
          f=f*j;
        cout<<"fact("<<i<<")="<<f<<"\n";
     }
 return 0;
}
```

اكتب برنامج يختبر فيما اذا كان العدد تام أم لا

العدد التام هو العدد الذي مجموع قواسمه يساويه

```
#include <iostream.h>
main()
{ int x,s;
  bool f;
  Cout<<"x="; Cin>>x;
  f=true;
  s=0;
  for(int i=1; i<x; i++)
     if( x % i==0)
        s=s+i;
     if( s!=x )
      f=false;
     if(f)   // if(f==true)
   Cout<<"this number is Complete";
     else
   Cout<<"this number isn't complete";
return 0;
}
```

اكتب برنامج يختبر فيما اذا كان العدد أولي أم لا (يقبل القسمة على نفسه والواحد فقط)

```
#include <iostream.h>
main()
{ int x,f;
  Cout<<"x="; Cin>>x;
  f=1;
  for(int i=2; i<x; i++)
     if(x%i==0)
       f=0;
   if(f==1)
  Cout<<"the number is Primary";
  else Cout<<"the number isn't Primary";
 return 0;
}
```

اكتب برنامج يختبر ما اذا كان العددين صديقين ام لا:

```
#include<iostream.h>
main()
{int x,y; Cin>>x>>y;
 int S1,S2;
 bool f;
 f=false; S1=0, S2=0;
 for(int i=1;i<x;i++)
   if(x%i==0)
     S1=S1+i;
 for(int j=1;j<y;j++)
   if(y%j==0)
     S2=S2+j;
 if ((S1==y)&&(S2==x))
   f=true;
 if(f)
   Cout<<"they are friends";
 else
   Cout<<"they are not friend";
 return 0;
}
```


اكتب برنامج يقوم بحساب مجموع مربعات الاعداد التامة ضمن المجال [1,x]

```
#include<iostream.h>
main
{ int x,S,S1;
 Cout<<"x:"; Cin>>x;
 for(int i=1;i<=x;i++)
 {  S1=0;
     for(int j=1;j<i;j++)
       if(i%j==0)
         S1=S1+j;
     if(S1==i)
       S=S+i*i;
 } Cout<<"S="<<S;
```

اكتب برنامج يقوم بحساب المتسلسلة

$$S = \frac{1!}{1+x} + \frac{2!}{2+x^2} + \cdots + \frac{n!}{n+x^n}$$

```
#include <iostreame.h>  #include <math.h>
main()
{ float x, s;
  int f, n;
  Cout<<"x="; Cin>>x;
  Cout<<"n="; Cin>>n;
  S=0; f=1;
  for( int i=1; i<=n; i++)
  {  f=f*i;
     S=S+f/(i+pow(x,i));
  }
  Cout<<"S="<<S;
  return 0;
}
```

اكتب برنامج يقوم بحساب مجموع الحدود الموجبة والسالبة للمتسلسلة:

$$S = +\frac{1!}{1+x} - \frac{2!}{2+x^2} + \frac{3!}{3+x^3}$$

```
#include <iostream.h>
#include <math.h>
main()
{ int n, f;
  float x, S1, S2, S;
  Cout<<"x="; Cin>>x;
  Cout<<"n="; Cin>>n;
  S1=0; S2=0; f=1;
  for(int i=1; i<=n; i++)
  { f=f*i;
     if(i%2==0)
       S1=S1+f/(i+pow(x,i));
     else
       S2=S2+f/(i+pow(x,i));
  }
  S=S2+S1;
  Cout<<"S="<<S;
  return 0;
}
```